AD-A233 328

MENTATION PAGE

Form Approved
OMB No. 0704-0188

1. Agency Use Only (Leave blank).	2. Report Date.	3. Report Type and Dates Covered.
	1990	Proceedings

4. Title and Subtitle.

Learning Automata: A Case Study

5. Funding Numbers.

Program Element No. APN
Project No.
Task No.
Accession No. DN25?017

6. Author(s).

Henry Rosche III and Maura C. Lohrenz

7. Performing Organization Name(s) and Address(es).

Naval Oceanographic and Atmospheric Research Laboratory
Ocean Science Directorate
Stennis Space Center, MS 39529-5004

8. Performing Organization Report Number.

PR 91:008:351

9. Sponsoring/Monitoring Agency Name(s) and Address(es).

Naval Air Systems Command
Department of the Navy (Air-805)
Washington, DC 20361-8080

10. Sponsoring/Monitoring Agency Report Number.

PR 91:008:351

11. Supplementary Notes.

DECUS

12a. Distribution/Availability Statement.

Approved for public release; distribution is unlimited.

12b. Distribution Code.

13. Abstract *(Maximum 200 words).*

Neural networks are trained to "learn" their expected behavior. Networks that are designed to learn a particular informational scheme are called learning critical to the operational performance of learning automata. In general, the automata is in some way "rewarded" for proper behavior and "punished" for wrong behavior. Initially, all choices of behavior are random, but by using these learning rules of reward an punishment that is used can significantly affect both local and global learning and results in surprising revelations about achieving proper behavior. Learning automata can be applied to a variety of computational problems. For example, a neural network can be trained to recognize which of several available filters, classifiers, or other neural networks are best suited to a particular task.

Scientists at the Naval Oceanographic and Atmospheric Research Laboratory's (NOARL's) Map Data Formatting Facility (MDFF) plan to apply this type of neural network training the their research in the automated feature extraction of digital maps. NOARL's dataset of interest consists of scanned aeronautical charts, provided by the Defense Mapping Agency, which are compressed by the MDFF computers into a form that is compatible with digital moving map systems on-board naval aircraft. In an effort to improve the quality of the output images, MDFF computer scientists are testing various digital image enhancement algorithms on this particular dataset. Learning automata could be used to help choose the best digital feature extraction process for a given subtask. For example, the vectorization of desert data requires a significantly different approach that that used to classify rugged terrain.

14. Subject Terms.	15. Number of Pages.
(U) Digital Maps; (U) Databases	4
	16. Price Code.

17. Security Classification of Report.	18. Security Classification of This Page.	19. Security Classification of Abstract.	20. Limitation of Abstract.
Unclassified	Unclassified	Unclassified	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

91 3 18 138

ARTIFICIAL INTELLIGENCE

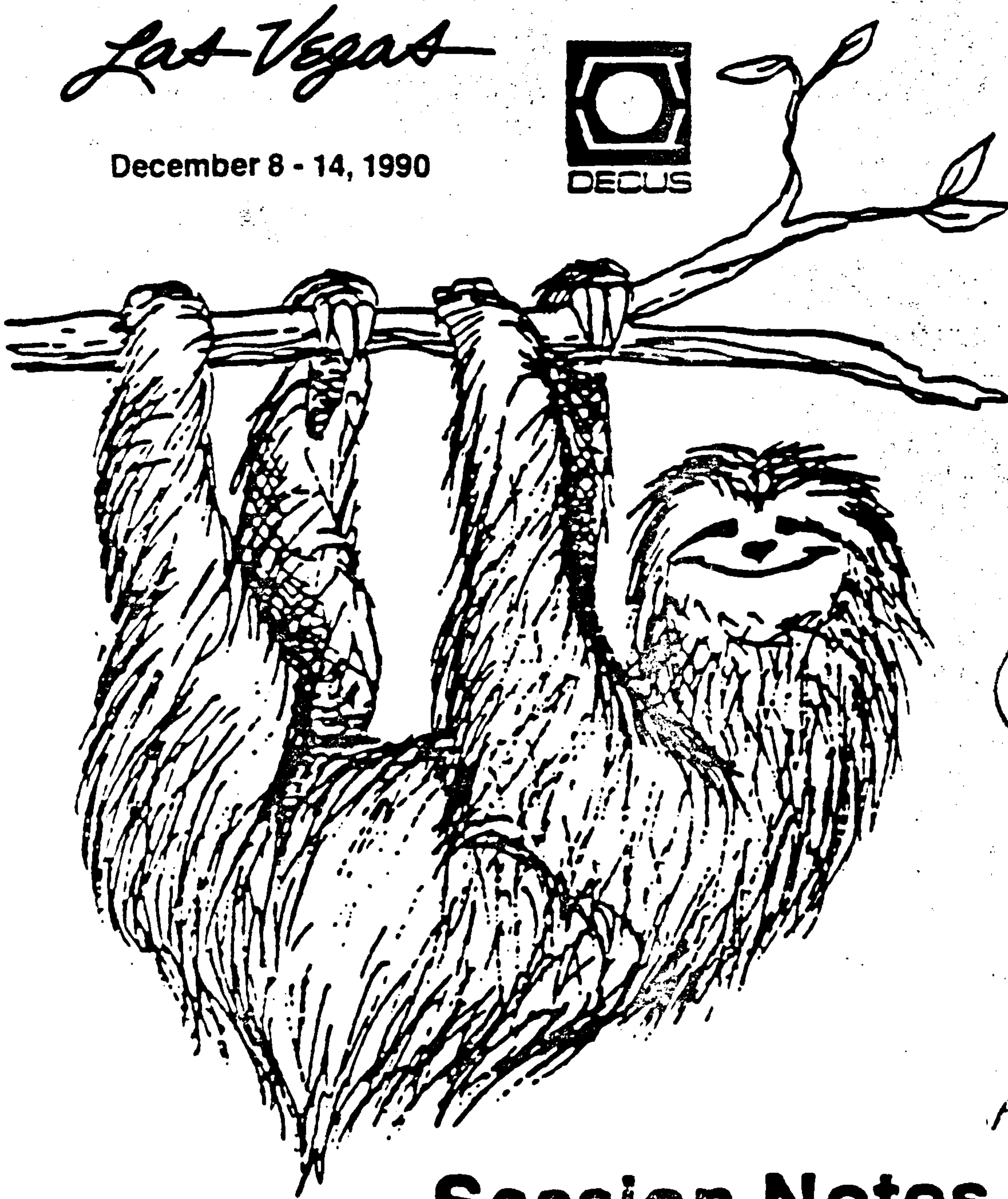
SPECIAL INTEREST GROUP

FALL DECUS U.S. Symposium

Las Vegas

December 8 - 14, 1990

DECUS



Session Notes

AI-4 A

LEARNING AUTOMATA: A CASE STUDY

Henry Rosche III and Maura C. Lohrenz

Naval Oceanographic and Atmospheric Research Laboratory
Mapping Sciences Branch
John C. Stennis Space Center, MS 39529

SESSION NOTES

Neural networks are trained to "learn" their expected behavior. Networks that are designed to learn a particular informational scheme are called learning automata. It is shown that the proper selection of training sets may be critical to the operational performance of learning automata. In general, the automata is in some way "rewarded" for proper behavior and "punished" for wrong behavior. Initially, all choices of behavior are random, but by using these learning rules of reward and punishment, proper behavior is eventually achieved. The amount of reward and punishment that is used can significantly affect both local and global learning and results in surprising revelations about achieving proper behavior. Learning automata can be applied to a variety of computational problems. For example, a neural network can be trained to recognize which of several available filters, classifiers, or other neural networks are best suited to a particular task.

15 Scientists at the Naval Oceanographic and Atmospheric Research Laboratory's (NOARL's) Map Data Formatting Facility (MDFF) plan to apply this type of neural network training to their research in the automated feature extraction of digital maps. NOARL's dataset of interest consists of scanned aeronautical charts, provided by the Defense Mapping Agency, which are compressed by the MDFF computers into a form that is compatible with digital moving map systems on-board naval aircraft. In an effort to improve the quality of the output images, MDFF computer scientists are testing various digital image enhancement algorithms on this particular dataset. Learning automata could be used to help choose the best digital feature extraction process for a given subtask. For example, the vectorization of desert data requires a significantly different approach than that used to classify rugged terrain.

This presentation describes the major results of an investigation into how best to teach an automata a desired behavior. An overview of these automata are given, and salient measures of performance are defined. The most significant viewgraphs from the presentation are attached and briefly described below.

1. Classic Student-Teacher Problem
The classic student-teacher relationship [1] is often studied in learning automata courses. In this relationship, the student begins by making random choices. Each choice is then sent to the teacher who rewards for a correct response or punishes for an incorrect one. The student receives the reward or punishment, updates his store of knowledge on the behavior, and makes another (and hopefully better) response.

2. A Cyclic State Transition Diagram
This state transition diagram shows the behavior to be learned. The automata starts in state one and attempts to learn how to proceed to state two. From state two it should continue to state three and so on. The entire cycle is learned in stages, not all at once. If the automata is in state one and chooses to go to state four (an incorrect choice for this model), it first would be punished for that choice, and then would attempt to learn where to continue from state four (in this case, back to state one). This is the basic mechanism that is used in the training loop.

3. The Connectivity Matrix
This connectivity matrix is used by the teacher to grade the student's responses. It contains the same information as the state transition diagram, but it is in a form that is easily stored in an array.

4. Variance of Similar Trials
1 The variance that occurs in three identical training trials [2] is shown. This variance is due to the early, random, choices of state change. A penalty of 1.0 (100%) means that all the statistical area stored for that change is taken from that change and distributed evenly to the other possible state changes. A reward of .95 means that 95% of each of the other possible state changes' information is taken from those states and given to this correct one. Note the random behavior in the first 10% to 15% of the training session. Also note that the trial run with the best learning (lowest curve) at the beginning of the trial has the highest curve (worst learned behavior) of all three at the end. Presumably, this indicates that in cases where more learning is required of the automata, more over-all improvement will result.

5. Averaging of Similar Trials
2 A significant reduction in erratic behavior is achieved by running each training session ten times and taking the average (mean) of all ten to represent that training session. Note that there is still some variance in the early stages before the three curves merge at the end of the training trials.

6. Different Reward and Penalty Measures
3 Different rates of learning are demonstrated with rewards and penalties that range from 99% to 9% in increments 10%. The highest curve represents 99% reward and 99% punishment, and each of the next lower curves represents the next lower level of reward and punishment. It is interesting to note that smaller rewards and puninshments result in quicker and better long-term learning than larger rewards and punishments.

7. Total Reward and Punishment vs. Partial Reward and Punishment
4 A scheme of 100% reward and 100% punishment (the top five curves) is contrasted with a scheme of 95% reward and 95% punishment (the bottom five curves). It is obvious that the 100% scheme results in no learning at all.

8. Pure Reward Versus Pure Punishment
5 The effect of having no reward with 95% penalty is represented by the top five curves. The bottom five curves depict the opposite: 95% reward with no penalty. Note that if a scheme must have either reward or punishment, then exclusive reward achieves better mid- and long-term learning than does exclusive punishment.

9. Values of Pure Reward
6 Two different levels of exclusive reward (no punishment) are shown: the top five curves represent 95% reward, and the bottom five curves represent 50% reward. In a scheme of pure reward, it is shown that smaller rewards create the desired behavior faster and more effectively than larger rewards.

A.I-4 B

REFERENCES

(1) K. S. Narendra and M. A. L. Thathachar, "Learning Automata - A Survey," in IEEE Transactions on Systems, Man, and Cybernetics, VOL. SMC-4, No. 4, July 1974.

(2) R. E. Walpole and R. H. Myers, "Probability and Statistics for Engineers and Scientists", 2nd ed., Macmillan Publishing Co., Inc. New York, 1978.

(3) H. R. Lewis and C. H. Papadimitriou, "Elements of the Theory of Computation," Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1981.

(4) R. S. Singleton and W. F. Tyndall, "Games and Programs," W. H. Freeman nad Co., San Francisco, 1974.

(5) P. H. Winston, "Artificial Intelligence," Addison-Wesley, Reading Mass., 1984.

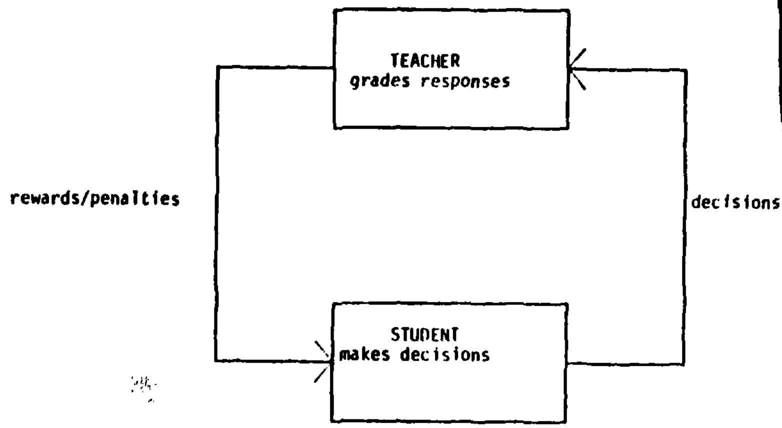
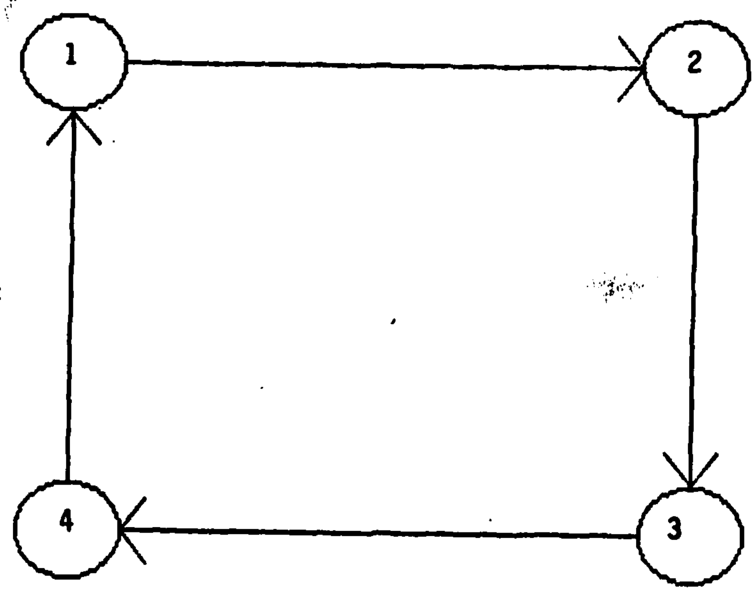


Diagram 1



State Transition Diagram

Diagram 2

$$
\begin{array}{c|cccc}
 & 1 & 2 & 3 & 4 \\
\hline
1 & 0 & 1 & 0 & 0 \\
2 & 0 & 0 & 1 & 0 \\
3 & 0 & 0 & 0 & 1 \\
4 & 1 & 0 & 0 & 0
\end{array}
$$

Connectivity Matrix

Diagram 3

AI-4
C

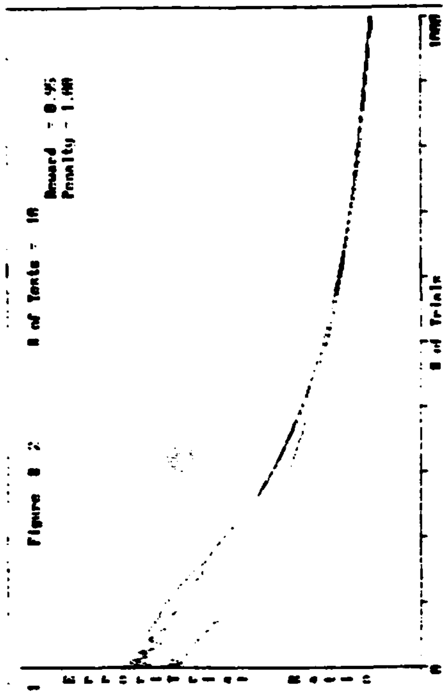


Figure
of Tests
Reward
Penalty
of Trials

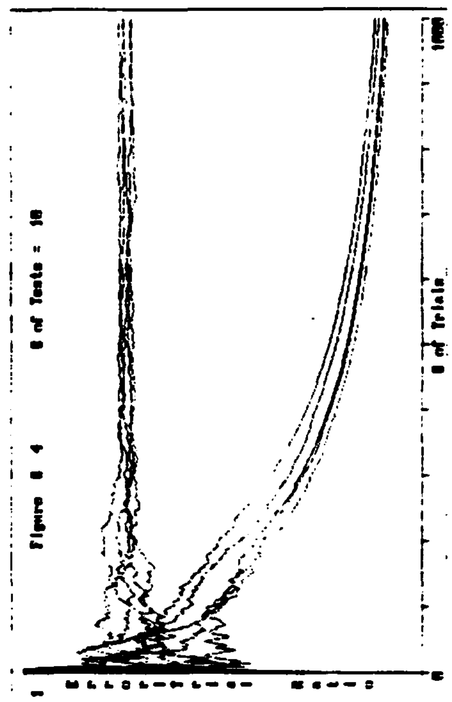


Figure
of Tests
of Trials

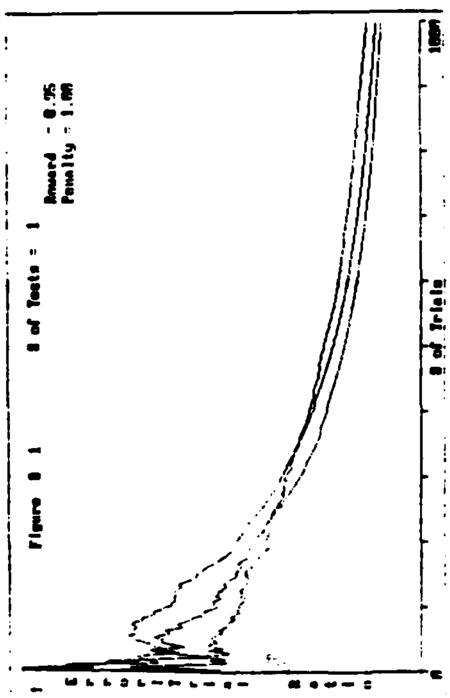


Figure
of Tests
Reward
Penalty
of Trials

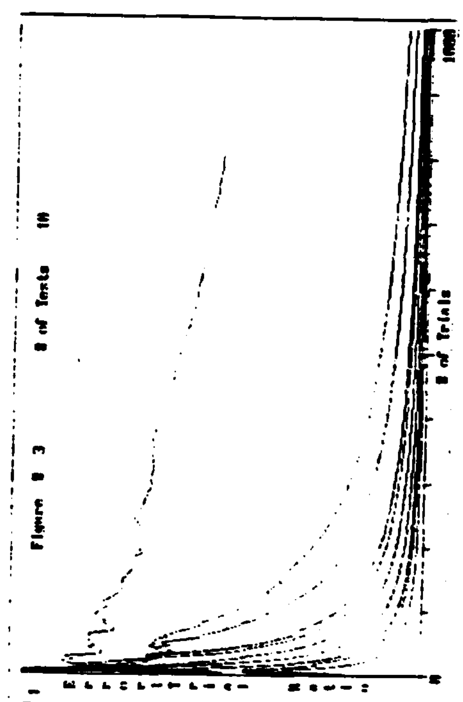


Figure
of Tests
of Trials

AI-4
D

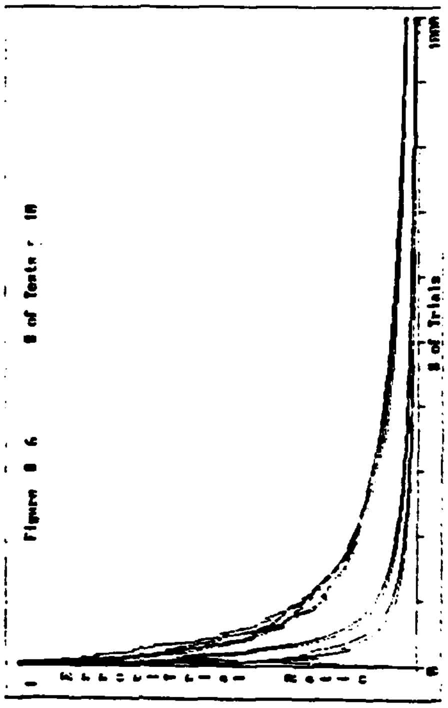


Figure B 6
# of Tests = 10
# of Trials
1000

Figure B 5
# of Tests = 10
# of Trials
1000